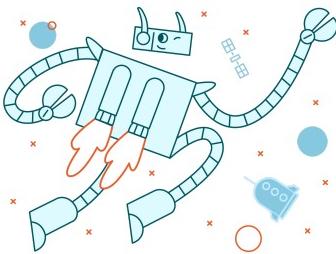


JavaScript Style Guide

Coding conventions are a set of guidelines that are recommended to follow when writing a code for a project. They give a uniform appearance to your code even if it is written by different people.



Why Are Coding Conventions Important

- Improve readability of code
- Improve maintainability of code
- Help detect errors easily



Variable & Function Names

- Use `camelCase` for variable and function names

Wrong:

```
let fruitcount = 0
let fruit_count = 0
let FruitCount = 0
```



Correct:

```
let fruitCount = 0
```

- All names should start with a letter

Wrong:

```
let 2array
```



Correct:

```
let toArray
```

Operators

We should always put spaces around operators, and after commas.

Wrong:

```
let x=y+z;  
const myArray=["apple", "orange" , "banana"]
```

Correct:

```
let x = y + z  
const myArray = ["apple", "orange", "banana"]
```

Code Indentation

We should always use 2 spaces for indentation of code blocks.

Wrong:

```
if (fruitCount === 0) {  
return "I have no fruits"  
}
```

Correct:

```
if (fruitCount === 0) {  
    return "I have no fruits"  
}
```

Statements

- Put the opening bracket at the end of the first line and use one space before it

Wrong:

```
if (fruitCount === 0){  
    return "I have no fruits"  
} else{  
    return "I have enough fruits"  
}
```

Wrong:

```
if (fruitCount === 0)  
{  
    return "I have no fruits"  
} else  
{  
    return "I have enough fruits"  
}
```

Correct:

```
if (fruitCount === 0) {  
    return "I have no fruits"  
} else {  
    return "I have enough fruits"  
}
```

- Put the closing bracket on a new line

Wrong:

```
while (fruitCount < 5) {  
    return "Buy more fruits"}
```

Correct:

```
while (fruitCount < 5) {  
    return "Buy more fruits"  
}
```

Line Length

We should avoid lines longer than 80 characters. If a statement does not fit on one line, break it after an operator or a comma.

```
document.getElementById("inputBox").value =  
"This is my new text"
```

